PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G09G 5/00

A1

(11) International Publication Number: WO 99/40562

(43) International Publication Date: 12 August 1999 (12.08.99)

(21) International Application Number: PCT/IL99/00083

(22) International Filing Date: 9 February 1999 (09.02.99)

(30) Priority Data:
09/020,812 9 February 1998 (09.02.98) US

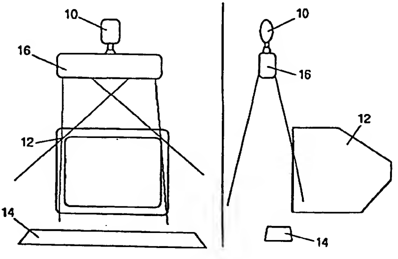
(71)(72) Applicants and Inventors: LEV, Joseph [IL/IL]; Sela Yaacov 2, 67537 Tel Aviv (IL). AZARIA, Haim [IL/IL]; Bnei Binyamin 28, 46362 Herzliya (IL).

(74) Agent: FRIEDMAN, Mark, M.; Beit Samueloff, Haomanim Street 7, 67897 Tel Aviv (IL).

(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, US, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: VIDEO CAMERA COMPUTER TOUCH SCREEN SYSTEM



(57) Abstract

A method and system for entering data into a computer via a computer monitor screen. A standard PC video camera (10) mounted above the computer screen (12) monitors the area immediately in front of the screen (12). A periscope-like optical system (16) located beneath the video camera (10) causes two images of the screen foreground to be recorded by the camera (10) simultaneously viewed from a different angle. Object recognition image processing is performed such as the user's finger (24, 26) or a pen. Spatial coordinates are generated and virtual space coordinates are then transformed into screen coordinates by means of linear interpolation and linear extrapolation from standard calibration points. In an alternative embodiment, only one image is recorded by camera (10), an object is identified by a spatial coordinate parameter and a perceived width parameter, then these parameters are transformed into screen coordinates by a calibration process.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

VIDEO CAMERA COMPUTER TOUCH SCREEN SYSTEM

FIELD AND BACKGROUND OF THE INVENTION

The present invention relates to a method for entering data into a computer and, in particular, it concerns a touch-screen data entry system.

It is known that several different methods can be used to facilitate data entry into a computer. Frequently, input of operational commands to a computer processor is achieved by means of a mouse or other pointing device. All pointing devices operate by effecting movement of a cursor on the display monitor in response to a comparable movement of the pointing device by the user. The user positions the screen cursor at a desired location on the screen (such as an item on a pull-down menu or a virtual "button" displayed on the screen) and then signals that the relevant operational command be implemented by "clicking" on the mouse.

As an alternative to physical pointing devices, video-image tracking techniques may be used to input operational commands to a processor. In these systems, video images of the user are acquired by a single video camera, and the images are processed so as to derive positional or other descriptive data about the user. This data is then translated into a specific operational command, or into a specific screen cursor location indicating a desired operational command. The user thus moves his hand, body, or a hand-held implement, while watching the computer screen, so as to activate desired operational commands.

Several such video-image tracking systems have been described. Thus U.S. Patent No. 5,767,842 to Korth et al, U.S. Patent No. 5,168,531 to Siegel, U.S. Patent No. 5,167,312 to Iura et al, and U.S. Patent No. 4,843,568 to Kreuger et al, all describe data input systems in which a single video camera is mounted either on the monitor and aimed at the user, or above the user and aimed down at the users hands. The position of the users hands or body in the acquired video image is described in terms of a set of two-dimensional XY coordinates, which are then translated into corresponding XY coordinates on the display monitor describing the location of the screen cursor. Alternatively, specific body gestures are recognized as corresponding to specific operational commands. In all these systems the video camera focuses on the user, who is located at some point distant from the display monitor.

As such, both pointing devices and standard video-image tracking systems are characterized by the phenomenon that the user performs a physical action, intended to implement an operational command, at a location distant from the focus of his attention, which of necessity is the computers display monitor rather than the users hand. This is in contrast to the way that a real control panel (such as a light switch, a telephone keypad, or the push-buttons of a microwave oven, as opposed to a virtual control panel depicted on a computer monitor) is activated, whereby the user extends his hand directly towards the focus of his attention, i.e. the control panel, and touches it so as to activate it. In this sense, the use of pointing devices or standard video-image tracking systems to activate a virtual control panel on a computer screen provide poor emulation of the natural process of control panel activation in real life.

In contrast to the above, computer touch-screen data input devices, in which the user manipulates his hand directly on the virtual control panel depicted on the computer screen so as to activate an operational command, allow for a natural and intuitive method of activating virtual control panels. Several such touch-screen data input methods have been described, all of which are characterized by the user manipulating his hand (or a hand-held implement) on, or immediately in front of, the computer screen.

Resistive methods utilize a low voltage current running through a resistive coating on the screen. When an object presses against the screen, the current flow, and thus voltage output, is altered. By monitoring changes in voltage, the location of a touching object is determined. In a similar manner, capacitive methods measure the change in capacitance of a screen caused by an object touching the screen, so as to determine the location at which the screen was touched.

Infrared methods utilize a network of infrared beams in front of the screen. A touching object disturbs this network, generating location data.

Surface-wave methods, as disclosed in U.S. Patent No. 5,591,945, send ultrasonic waves through a specialized coating on the surface of the screen. An object touching the screen disrupts the ultrasonic waveform and generates location data.

Force-sensor methods, as disclosed in U.S. Patent No. 5,541,372, utilize force activated sensors on the computer screen to measure deformation of the screen when it is touched by an object.

The above touch-screen methods, however, suffer from several deficiencies, as

follows:

1. None of these technologies are able to discriminate between multiple simultaneous touches, and they thus allow for only a single screen touch at any one moment in time.
2. All of these technologies utilize dedicated hardware which is built into or around the particular screen being used. As such, these systems are dedicated to a particular display screen, and generally cannot easily be transferred from one computer display screen to another. Furthermore, once installed, these systems cannot easily be adapted to a screen of different size to that of the screen on which the system was first installed.
3. The specialized coatings and hardware utilized in resistive, capacitive and surface-wave systems all disturb the transmission of free light from the display screen, thus degrading the quality of the display image as viewed by the user.
4. Capacitive systems require frequent calibration. In addition, an electrically isolated object (such as a pen or a glove) cannot be sensed when touching the screen.
5. Infra-red systems can only be implemented on flat screens, and suffer from low resolution.
6. In addition to "location", many other attributes describe the object used to touch a computer screen. These additional attributes, such as the size, orientation, distance from the screen, and color of the object, could themselves be utilized to convey data to the computer. All of the above described systems, however, are only capable of sensing the location of an object as it touches the screen.

To date, it has not been feasible to utilize video-image tracking technology, which does not suffer from the deficiencies of non-video based touch-screen systems as mentioned above, to implement touch-screen data input systems. This is because direct video imaging of a display screen often results in graphic ambiguity and interference with image processing functions. Consequently, for video-image tracking to be effective the acquired images must exclude images of the display screen. As such, the proximity of the users hand to a virtual control panel on the display screen is

not inferable from the acquired graphic video data. As activation of operational commands in touch-screen systems is triggered by the users hand reaching a critical proximity to, or actual touch of, the display screen, it has not been feasible to achieve a true touch-screen data input system based on current video-image tracking techniques.

In dual-video tracking systems two or more video cameras are used to acquire simultaneous images of a scene from two or more different viewpoints, as opposed to the single viewpoint acquired by the single video camera in the video-image tracking systems described herein above. Processing of the images acquired by dual-video tracking systems allows the spatial locations of objects within the imaged scene to be defined in terms of three orthogonal axes (X, Y, and Z). This is in contrast to the two dimensional localization of imaged objects achievable by single-camera video-image tracking systems. Dual-video tracking methods have been used in systems designed to render three dimensional graphic depictions of imaged scenes, or to construct virtual reality bason real-life scenes. Azarbayejani et al have described a dual-video tracking system for recovering three-dimensional descriptions of humans from images in real time (Azarbayejani A, Wren C, Pentland A: Real-Time 3-D Tracking of the Human Body. In: Proceedings of IMAGE'COM 96, Bordeaux, France, May 1996, and reported in M.I.T. Media Laboratory Perceptual Computing Section Technical Report No. 374). The described applications of this system relate primarily to depictions of virtual realities, avatars and telepresence, visually guided animation, and sign language recognition. In addition, the system described by Azarbayejani et al. can be used to transmit operational commands to a computer processor in a manner similar to that described above for single-camera video-image tracking systems, namely, by utilizing gesture or body position recognition. Thus in all single and dual video tracking systems described to date, the cameras focus on the user at a location distant from the display monitor. such that the implementation of operational commands is achieved according to the paradigm of a pointing device rather than that of a touch screen.

There is thus a widely recognized need for, and it would be highly advantageous to have, a computer touch-screen data entry system which is able to process multiple simultaneous touches, can easily be transferred from one computer display screen to another, can easily be adapted to screens of different sizes, does not

mechanism" is defined herein as referring to a combination of any type of video capture system with any type of optical mechanism in such a manner as to result in the capture of more than one simultaneous sub-image of a scene, as well as referring to a combination of video capture mechanisms capable of capturing more than one simultaneous sub-image of a scene without the use of an additional optical system. In alternative embodiments of the current invention, therefore, a multi-image video capture mechanism may include multiple video cameras which feed simultaneous captured images of the same scene into a processor, or may include a combination of a video camera with a fiber-optic mechanism for generating multiple simultaneous images of the same scene.

It is the generation of multiple simultaneous images of a screen foreground, each from a different viewpoint of that screen foreground, that facilitates the extraction of three dimensional data about objects located in the screen foreground. As opposed to prior art video-image tracking systems which focus on the user at a location distant from the display monitor, the system of the current invention focuses on the immediate screen foreground, from a perspective oriented along, and substantially parallel to, the XY plane of the screen, rather than focusing on the user. Thus as the user extends his hand towards a virtual control panel depicted on the computer screen, his hand enters the scene being imaged by the video system. Three dimensional coordinates describing the location of the user's hand in space are then derived from the stereo video images. As the imaged scene is immediately adjacent to the screen itself, the screen functionally constitutes one of the margins of the image, defining abscissa of the Z-axis (which is the axis running orthogonally to the plane of the screen, extending from the screen towards the user) of the imaged scene. A Z-axis displacement is predefined as being the critical proximity to the screen which, when attained by the users hand, activates the operational command represented by the virtual control button (in the XY plane) on the touch screen. By utilizing the Z axis displacement of the user's hand (or a hand held implement) relative to the screen, rather than gesture recognition, a functional touch-screen data input system is achieved based on dual-video image tracking techniques.

In a preferred embodiment, a colored background material placed beneath the screen foreground enhances the image definition of any objects, such as the user's finger or a pen, within the screen foreground. The two images are processed by

specialized software within the computer, to define a unidimensional location for each object in each of the two images, or to define other attributes of the objects such as their colors, shapes, distances from the screen, or orientations in space. The actual locations of the objects relative to the screen are then calculated from the defined unidimensional locations. The locations of the objects, or the other defined attributes of the objects, serve as data inputs for the computer. Alternatively, colored background material is not utilized, and image-processing techniques utilizing disparity maps are employed to differentiate the user's hand from the background.

In a second preferred embodiment, a periscope-like optical system is not utilized, such that only one video image of the screen foreground is captured. An object (such as a pointer or the user's finger), the width of which has been previously calibrated, is introduced into the screen foreground. The acquired image is processed by specialized software within the computer to define a spatial coordinate of the object relative to the screen, and to define a perceived width of the object. The latter two parameters are then transformed into screen coordinates by utilizing standardized conversion factors and constants derived from a previous calibration process.

As the device of the present invention is mounted externally to the computer monitor, and not integrated into the screen itself, it is easily installed on screens of different shapes and sizes, and can be moved from one screen to another. The video camera and periscope optical system is thus simply added on to any existing computer screen, and the image processisoftware installed in the computer. The PC video camera used in the device is a standard, non-dedicated camera, which is already a component of many computer systems. There is no need for any other add-on hardware other than the camera and the periscope optical system.

According to the teachings of the present invention there is therefore provided a system for entering data into a computer by interacting with a screen having a foreground, including a video capture mechanism, operative to capture at least one image of the screen foreground; and an image processor, operative to identify at least one object within the image, measure at least one descriptor of the identified object, and transform the at least one descriptor into a screen coordinate. There is also described a method for entering data into a computer by interacting with a screen, including the steps of positioning an object in a foreground of the screen, acquiring at least one image of the screen foreground, processing each of the at least one acquired

image to identify a first object in the screen foreground, inferring at least one descriptor of the object in each of the processed images, each of the at least one descriptor being a coordinate of a point in a virtual space. and effecting a transformation of the virtual coordinates into screen coordinates describing a location of a point on the screen. There is further described a system for entering data into a computer by interacting with a screen having a foreground, including a video capture mechanism, operative to capture a plurality of simultaneous images of a screen foreground, each of the simultaneous images depicting the screen foreground from a different viewpoint ; and an image processor, operative to identify at least one object within the image, measure at least one descriptor of the identified object, and transform the at least one descriptor into a screen coordinate. There is also described a method for entering data into a computer by interacting with a screen, comprising the steps of positioning an object in a foreground of the screen; acquiring a plurality of images of the screen foreground, each of the images depicting the screen foreground from a different viewpoint ; processing each of the plurality of acquired images to identifya first object in the screen foreground; inferring at least one descriptor of the object in each of the processed images, each of the at least one descriptor being a coordinate of a point in a virtual space; and effecting a transformation of the virtual coordinates into screen coordinates describing a location of a point on the screen.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention is herein described, by way of example only, with reference to the accompanying drawings, wherein:

FIG. 1 is a schematic depiction, from the front and the side, of the hardware configuration of the present invention, showing typical locations of the video camera, optical periscope system, and blue background material, in relation to a computer monitor;

FIG. 2 is a diagram illustrating the functioning of the optical periscope system;

FIG. 3 is an example of a typical image of a screen foreground as captured by the video camera (FIG 3a), and the same image after image processing to identify objects in the screen foreground (FIG. 3b);

FIG. 4 is a graphic depiction of points mapped in the LRV (Left Right Views) space, and their corresponding XY locations on the display screen;

FIG. 5 is a graphic depiction of points mapped in the LRV (Left Right Views) space, and their corresponding XY locations on the display screen, showing the location of calibration points;

FIG. 6 is a graphic depiction of points mapped in the LRV (Left Right Views) space, showing calibration points and calibration areas;

FIG. 7 is a schematic depiction of the hardware configuration of a second preferred embodiment of the present invention; and

FIG. 8 is a graphic depiction of points mapped in the PW (Position-Width) space.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

The present invention is of a computer touch-screen data input system and method.

Before explaining at least one embodiment of the invention in detail, it is to be understood that the invention is not limited in its application to the details of construction and the arrangement of the components set forth in the following description or illustrated in the drawings. The invention is capable of other embodiments or of being practiced or carried out in various ways. Also, it is to be understood that the phraseology and terminology employed herein is for the purpose of description and should not be regarded as limiting.

The principles and operation of a computer touch-screen data input system, according to the present invention, may be better understood with reference to the drawings and the accompanying description.

Referring now to the drawings, FIG. 1 schematically depicts the hardware configuration of a first preferred embodiment of the present invention, as seen from the front and the side. The hardware components of this embodiment are a standard PC video capture system **10** (such as a PC video camera), a mounting arm to hold video capture system **10**, an optical system **16** including appropriately mounted reflectors, and an optional sheet of colored material **14**. As shown in FIG. 1, video capture system **10** is located above computer display **12**, looking vertically down. In

alternative embodiments, video capture system 10 may be located at any other location relative to computer display 12 (such as to the side), provided that the location of video capture system 10 allows for the capture of an image of the foreground of computer display 12. Optical system 16 is optically coupled to the lens of video capture system 10. Sheet 14 is located immediately beneath and in front of computer display 12, such that sheet 14 forms a background for any objects in the screen foreground of computer display 12. In the preferred embodiment, sheet 14 is a sheet of blue plastic or paper approximately 80 cm. in length and 8 cm. in width. For a normal home PC configuration, sheet 14 is placed on the working surface on which the computer stands, between computer display 12 and the computer keyboard.

FIG. 2 illustrates the functioning of optical system 16. In a preferred embodiment, optical system 16 includes two pairs of reflectors, such as prisms or mirrors, such that each pair of reflectors forms a periscope. Hereinafter, the term "periscope" refers to any optical system which allows a scene to be viewed from a viewpoint different from the viewpoint at which video capture system 10 is located. One pair of reflectors, forming a first periscope 18, projects an image onto the upper half of the image captured by video capture system 10. This projected image is the view that video capture system 10 would see if it were shifted approximately 10 cm, ranging from 5-15 cm, to its left. First periscope 18 thus simulates a virtual camera 20 which views the screen foreground from a left-shifted viewpoint. Similarly, a second pair of reflectors, forming a second periscope 22, projects a right-shifted image onto the lower half of the image captured by video capture system 10, thus simulating a second shifted virtual camera 24.

In an alternative embodiment, optical system 16 contains only one periscope, which generates a shifted image in video capture system 10. In this embodiment, the second image captured by video capture system 10 is the non-shifted image seen directly by the video camera. Optical system 16 thus combines two views of the screen foreground into one image, with the upper half of the image containing the left-shifted view, and the lower half of the image containing the right-shifted view, or vice-versa (or, in an alternative embodiment, one half of the image containing a non-shifted view).

Video capture system 10 preferably is any commercially available PC video capture system that preferably meets the following specifications:

1. The system can capture color images in RGB format. This is important because the objects which will be identified in the screen foreground will be contrasted against a background. In the preferred embodiment, the system can capture color images in 24-bit RGB format.
2. The system can supply a frame rate of more than 3 frames per second, and preferably about 25 frames per second. This is necessary to ensure that the software monitoring process (described below) does not skip some touch events.

An example of a digital video capture system suitable for use in the present invention is a Philips PC Camera (Video Camera Modules/Philips Business Electronics, Eindhoven, The Netherlands). In an alternative embodiment, in which optional sheet **14** is not used, a PC video camera which does not capture color images may be used as video capture system **10**.

As no hardware is located immediately in front of the display screen, which would disrupt the users line of sight, no degradation of the quality of the display image occurs.

The software component of the current invention controls the operation of video capture system **10** and analyzes the captured images in real time, so as to identify and localize objects within the screen foreground. Any object located within the screen foreground and imaged by video capture system **10** is hereinafter also referred to as a "perceived object". In the preferred embodiment, the software component of the current invention is located within the processor of the computer with which the current touch-screen data entry system is being used. In an alternative embodiment, the software component may be located externally to the computer with which the current touch-screen data entry system is being used.

The functioning of the preferred embodiment of the current invention is detailed below.

The user introduces an object into the screen foreground, for the purpose of pointing at or touching computer display **12**. Generally, the object is a pen, a pointer, or the user's hand or finger, however any object may be used provided it does not have a color on it which is similar to the background color of sheet **14**. Video capture system **10** captures images of the object as it enters and moves through the screen

foreground. As explained above, each captured image contains two sub-images of the screen foreground, as seen from the perspectives of virtual cameras 20 and 24: an upper sub-image 30 which represents the left-shifted viewpoint, and a lower sub-image 32, which represents the right-shifted viewpoint, as shown in FIG. 3a.

The software component of the preferred embodiment of the current invention performs real time simple object recognition. In both sub-images 30 and 32, sheet 14 can be seen in the background, with two fingers 24 and 26 approaching the display screen. The process of object recognition as performed by the object recognition software is as follows:

1) The first step is to separate each captured image into it's component upper and lower sub-images.

2) The second step is to identify objects crossing the background. As such objects obscure sheet 14, they can be referred to as "obscuring objects". This process of identification is achieved by separating the background blue of sheet 14 (in the preferred embodiment) from the non-blue color of obscuring objects. Therefore, for each pixel of each sub-image, the three color values (the Red Green Blue numbers) for the pixel are examined, and a predefined three dimensional decision table is consulted, so as to determine if the pixel is of the same blue color as sheet 14. If the pixel is the same color as sheet 14, the pixel is designated as being "blue" (i.e. corresponding to the background of the screen foreground). If it is not, the pixel is designated as being "non-blue"(i.e. corresponding to an object in the screen foreground). The result of this analysis is a processed image containing either blue or non-blue (e.g. white) pixels, as shown in FIG. 3b. It will be understood that the same process can be performed using any background color, in addition to blue. Multiple obscuring objects can be identified simultaneously in this manner, provided that they do not overlap one another.

3) The processed image is then analyzed by sequentially examining each row of pixels to identify the occurrence of adjacent white pixels lying between surrounding blue pixels, forming a horizontal "run" of white pixels. By "horizontal" is meant an orientation which is approximately

parallel to that of the surface of computer display 12, as seen in the relevant sub-image. Horizontal runs in neighboring rows that are touching each other are grouped together, and are taken to represent an obscuring object (such as finger 24 or 26) in the screen foreground. For each identified horizontal white run, the center of the run is marked. The center markings of a group of white runs constitute a vertical skeleton 28 of the object. By "vertical" is meant an orientation which is approximately perpendicular to that of the surface of computer display 12, as seen in the relevant sub-image. The pixels that belong to object skeleton 28 of an obscuring object provide a rough estimation of the direction in which the obscuring object is pointing.

4) The next step is to determine the general direction of each object and the horizontal position of each object on the image. For each object skeleton 28, the single pixel closest to the surface of computer display 12 could theoretically mark the touching edge of an obscuring object. However, due to the statistical noise inherent to video images, this pixel is an unreliable indicator of the true touching edge of an obscuring object. Therefore, linear regression analysis of the pixels of object skeleton 28 is used to determine a straight line passing through the object. The intersection of this straight line with the horizontal white run closest to the surface of computer display 12, i.e. the white run at the edge of the object, is designated as the object's "touch point".

5) The following step is to match the images of the same obscuring object, as seen in sub-images 30 and 32, with each other. A list of object touch points, defined by their locations on the horizontal axis, is generated for each sub-image 30 and 32. As the locations of the object touch points are defined only in terms of the horizontal axis of the image, each entry in this list is said to be a "unidimensional" location of a touch point. The two lists of unidimensional locations are then merged into one list according to the horizontal order of the objects found in each list, resulting in a combined list where each object touch point is designated with two numbers: a horizontal axis location on left-shifted sub-image 30, and another horizontal axis location on right-shifted sub-image 32.

Each entry in this combined list is therefore a coordinate value defining the location of a touch point. Furthermore, these coordinates are said to constitute a two-dimensional virtual space defining the location of the obscuring object(s) in the screen foreground. Each set of two coordinates thus describes a location in this two dimensional virtual space, hereinafter referred to as the "LRV space" (Left Right Views space).

6) An additional, optional, step is to extract data describing additional attributes of each obscuring object, such as the object's color, width, and direction. Thus, the colors of the obscuring pixels (i.e. the pixels belonging to the obscuring object) are averaged (using the RGB data in the original, full color, captured image), so as to describe the average color of the obscuring object. Additionally, a width attribute is generated for each obscuring object by averaging the lengths of the white runs of that object, and compensating for the spatial orientation, position, and direction of the object, all of which may affect the viewed width of the object. By determining the width attribute of an obscuring object, different types of objects can be differentiated from each other (for example, a fist can be differentiated from a finger). The distance of the obscuring object from the screen is calculated from the location of the object's touch point relative to that of the surface of computer display **12**. Finally, the angle of the straight line describing object skeleton **28** is calculated. Thangle describes the direction in which the obscuring object is pointing (for example, a finger pointing from left to right versus one pointing from right to left), and is thus an additional useful attribute of the object.

7) After identification of the objects in a single image is completed, the image is compared with the previous image, so as to determine whether objects have appeared, moved, or disappeared.

Once the process of object recognition has been completed, the location of the obscuring object in the virtual LRV space is transformed into a location (defined by X and Y coordinates) on the screen of computer display **12** by means of a mathematical transformation which will be explained in reference to Figures 4, 5, and 6 below. The

X and Y coordinates defining the location of any point on the screen of computer display **12** are hereinafter also referred to as "screen coordinates".

As explained above, each perceived obscuring object is described in terms of two coordinates, one specifying its horizontal axis position in sub-image **30** (hereinafter referred to as value V1), and the other specifying its horizontal axis position in sub-image **32** (hereinafter referred to as value V2).

FIG. 4 shows the result of an experiment in which a single object, located in a screen foreground and viewed by a dual-image optical system as described above, was moved along a computer screen so as to trace a set of straight horizontal and vertical lines, thus forming a grid on the screen surface. The acquired images were processed to generate a set of V1 and V2 values, as described above. The V1 and V2 values were then plotted against each other, resulting in the graph shown on the left side of FIG. 4. Each point on this graph represents a single touch along the path traced by the object along the screen surface. The extreme screen points, which are the Left-Top, Right-Top, Left-Bottom, and Right-Bottom corners of the screen, are marked on the graph with labels **LT, RT, LB,** and **RB** respectively. This graph is thus a representation of the LRV space. On the graph an imaginary continuation of the vertical screen grid lines (continuing down below the bottom of the screen) is shown. As these lines are parallel, they meet at infinity, marked by the point labeled **Inf.** on the "LRV space" graph. As the path traced by the object on the computer screen is known, it will be understood that each point depicted on the LRV space graph of FIG. 4 can be matched with a corresponding point on the surface of the computer screen. These corresponding points are shown in the grid on the right side of FIG. 4, which depicts the XY screen coordinates of the path traced by the object on the computer screen. Any point on the "LRV space" graph can therefore be translated into it's corresponding XY coordinate on the computer screen (i.e. a screen coordinate), once the calibration between the two coordinate sets is known.

Therefore, each time the device of the current invention is installed on a computer monitor, or its position on a computer monitor is altered, a manual calibration procedure is performed.

The calibration procedure is performed as follows:

1. video capture system **10** is activated.

2. Eight predefined points, hereinafter also referred to as "standard

points on the screen", are shown to the user on the computer screen. and the user is asked to touch each point, one at a time. The V1 and V2 values for each point touched by the user are recorded. Thus, at the end of the recording process 8 sets have been generated. with 4 numbers in each set (X_i, Y_i, $V1_i$, and $V2_i$, where "i" runs from 1 to 8 and identifies each of the eight calibration points touched on the screen, and X and Y represent the actual coordinates of the calibration points on the computer screen surface). The locations of the eight calibration points are predefined such that they cover most of the screen, as shown by the 8 dots (labeled 41 to 48) on FIG. 5. The software application then automatically calculates the calibration between the coordinates in each of the eight sets.

3. On the LRV graph "Vertical Lines" (by which is meant lines that, on the computer screen XY space, are vertical) are constructed. These lines connect points 41 to 45, 42 to 46, 43 to 47, and 44 to 48. Similarly, six "horizontal lines" are constructed connecting points 41 to 42, 42 to 43, 43 to 44, 45 to 46, 46 to 47, and 47 to 48. These straight lines (shown in FIG. 6) thus divide the LRV space into several areas.

In an alternative embodiment, a calibration procedure is performed automatically (as opposed to the manual procedure described above). In this alternative embodiment, the computer screen displays several white dots against a black background, the white dots being positioned at predefined locations on the screen. Video capture system 10 captures two simultaneous images of the computer screen. The captured images are then image processed to identify the white dots, and, by a procedure analogous to that described above for manual calibration, the screen coordinates for each white dot are correlated with the LRV virtual space coordinates generated from the two images.

As this calibration is easily and rapidly performed, and as the hardware of the present invention is located externally to the computer touch-screen (and is thus easily mountable and removable) the system of the current invention can easily be transferred from one computer to another, and can easily be adapted to screens of different sizes.

In a further alternative embodiment of the current invention, sheet 14 is not included in the system. In terms of this embodiment, objects (implements or the users hand) introduced into the screen foreground are not identified and localized by means of analyzing their color characteristics, as contrasted against a specific background color (such as blue). Rather, the coordinates of a screen foreground object in the LRV space are derived as follows:

For each pixel in sub-image 30 a single matching pixel in sub-image 32 is identified, by building a "disparity map". This process is facilitated by checking matches along epipolar lines, and can be achieved for most pixels. The processes of generating disparity maps and checking matches along epipolar lines have been well described in the prior art (Milan Sonka, Vaclav Hlavac, and Roger Boyle: Processing, Analysis, and Machine Vision, 2nd Edition, published by PWS - an Imprint of Brooks and Cole Publishing, 1998, ISBN 0-534-95393-X) which is incorporated herein by reference. Pixels in subimage 30 which represent an object that is obscured in sub-image 32, or vice-versa, are ignored. Each pair of pixels is then mapped into a screen foreground XYZ position using projective transformations. The pixels that come from the background are separated from objects in the screen foreground by noticing that their Y coordinates (indicative of the height of the object above the background) are significantly below the screen bottom. These "background" pixels are then discarded. Thus, in this embodiment, the software builds an internal three-dimensional model of the imaged objects using algorithms well known in the art for analyzing stereo images. These algorithms use information gathered from the two viewpoints to calculate the XYZ coordinate of each pixel that is seen in both images. This three-dimensional information is used to separate the background from moving objects on top of that background. The background may be the desk on which the computer screen stands, while the moving objects may be the user's hands, or tools that operate in the screen foreground in relation to objects and images depicted on the screen. If there is a need to image objects that are hidden by other objects, additional stereo cameras are added at different places.

In this embodiment, a self-calibration process is implemented using techniques well known in the art wherein the user waves one finger in front of the camera while the software learns about matching points in the two views.

The mathematical transformation used to translate a point on the LRV space

into it's corresponding XY coordinate on the computer screen is as foll:

1. The LRV point is first classified as falling within one of the areas into which the LRV space was divided during the calibration process.
2. If the LRV point is within an area surrounded by four calibration points, linear interpolation using the LRV coordinates and the four calibration coordinates is performed, to calculate the corresponding XY coordinate on the computer screen. If the LRV point is within an area bordered by only two calibration points (i.e. an area on the periphery of the LRV space), linear extrapolation using the neighboring lines and the relevant LRV and calibration coordinates is performed, to calculate the corresponding XY coordinate on the computer screen.

The shortest distance between the touch-point and the surface of computer display 12 is then measured in the "Z axis" of the computer screen. If the touch-point is found to be within a maximum predefined distance from the surface of the screen (for example, 2 cm), the touch point is defined as "touching" the screen. When the touch-point is defined as touching the screen, the computer screen XY coordinate which was calculated as described above is input to the host computer, thus completing the process of touch-screen data input.

In an alternative embodiment, the same principles as described above for defining a two-dimensional XY coordinate within the screen foreground can be used to define a three-dimensional XYZ coordinate within the screen foreground. This is achieved by acquiring and processing three or more simultaneous sub-images of the screen foreground, describing unidimensional locations of a perceived object in each sub-image, combining the unidimensional values into a multidimensional coordinate describing the location of the perceived object in a three-dimensional virtual space, and then transforming the virtual space coordinates into three-dimensional screen foreground XYZ coordinates for the perceived object.

FIG. 7 is a schematic depiction of a second preferred embodiment of the present invention. The hardware components of this embodiment are standard PC video capture system 10 (as described above for the first preferred embodiment), a mounting arm to hold video capture system 10 (not shown), and an optional sheet of

colored material 14. As in the first preferred embodiment, video capture system 10 is located above computer display 12, looking vertically down. Note that in this embodiment, as compared to the first preferred embodiment, optical system 16 is not utilized.

The functioning of this second preferred embodiment is as follows.

The user introduces an object 60 into the screen foreground, for the purpose of pointing at or touching computer display 12. Object 60 is any object which can be used as a pointer (such as a pen, a pointer, a ruler, or the users hand or finger), and which was used to perform an initial calibration procedure, the details of which are described below. Video capture system 10 captures images of object 60 as it enters and moves through the screen foreground. Each of these captured images is then processed by the object recognition software, which performs real time simple object recognition as described below:

1) First, object 60 crossing the background is identified. This process of identification is achieved in a manner identical to that described above for the first preferred embodiment. Thus, the background blue of sheet 14 is separated from the non-blue color of object 60, to produce a processed image containing either blue or non-blue (i.e. white) pixels. The processed image is then analyzed by sequentially examining each row of pixels to identify the occurrence of adjacent white pixels lying between surrounding blue pixels, forming horizontal "runs" of white pixels. Horizontal runs in neighboring rows that are touching each other are grouped together, and are taken to represent object 60.

2) A vertical skeleton 28, and a touchpoint, of object 60 are then marked, in a manner identical to that described above for the first preferred embodiment.

3) The unidimensional location of the touchpoint on the horizontal axis (P) is then defined.

4) The width of object 60, as perceived in the processed image, is then calculated from the lengths of the horizontal runs of white pixels in the processed image. As this measured width is not the actual width of object 60, but is rather the width of the image of object 60, this measured width can be described as being a "perceived width". By

"perceived width" is meant the width of an image of an object. rather than the true width of the object itself. It will be understood that the perceived width of an object is determined by two factors (assuming that no magnification or reduction of the image occurs): the actual width of the object, and the distance (D) of the object from the camera generating the image being measured. As the object approaches the camera, therefore, its perceived width approaches that of the actual width of the object. Conversely, as the object becomes more distant from the camera, its perceived width diminishes, approaching zero as the object approaches infinity. A "Perceived Width" value (W) for object 60 is thus obtained, this Perceived Width value being an expression of the unidimensional location of object 60 on the Z axis (that is, the axis running towards or away from video capture system 10) of the screen foreground.

The unidimensional location of object 60 on the Z axis (the Perceived Width value- W) is then combined with the unidimensional location of the touchpoint on the horizontal axis (P) to give a coordinate set defining a point in a two-dimensional virtual space, hereinafter called the "PW" (Position-Width) space. The PW space is thus a polar coordinate space, akin to the LRV space described above for the first preferred embodiment.

FIG. 8 is a graphical depiction of the PW space. The graph shown in the figure shows the result of an experiment in which a single object was located in a screen foreground and viewed by a video imaging system as described above. The object was moved along a computer screen so as to trace a set of straight horizontal and vertical lines, thus forming a grid on the screen surface. The acquired images were processed to generate a set of P and W values, and the generated values were plotted against each other, resulting in the graph shown. The coordinates corresponding to the left top, right top, left bottom and right bottom corners of the screen are marked as **LT**, **RT**, **LB**, and **RB** respectively. Also marked is the point at which extensions of the left and right screen borders meet. As the screen borders are parallel, this point is at infinity, and appropriately corresponds to a Perceived Width value of zero.

The location of object 60 in the virtual PW space is then transformed into a location (defined by X and Y coordinates) on the screen of computer display 12 by

mapping polar coordinates from the PW space to the XY space, using the following formulae:

i) $D = K / W$

ii) $\theta = a * P + b$

iii) $X = Xc + D * \sin(\theta)$

iv) $Y = Yc + D * \cos(\theta)$

where:

"P" is the measured horizontal position of an object in the screen foreground, in pixels;

"W" is the width of that object in pixels, as measured on an acquired video image;

"D" is the distance between the object and the camera, measured in screen pixel units;

"K" is a constant describing the transformation between W and D;

θ is the angle, in radians, between the axis of the camera lens and the location of the object, when the axis of the camera lens is assumed to be aligned with the center of the screen foreground (such that θ is zero when the object is at the center of the screen foreground);

"X" is the screen X-axis coordinate of the object , in screen pixel units;

"Xc" is the screen X-axis coordinate of the camera (obtained by extrapolation from the screen XY grid) in screen pixel units;

"Y" is the screen Y-axis coordinate of the object , in screen pixel units;

"Yc" is the screen Y-axis coordinate of the camera (obtained by extrapolation from the screen XY grid) in screen pixel units;

"a" is a conversion factor describing the linear transformation between P (measured ipixels) and θ (measured in

radians); and

"b" is a constant describing the angle (in radians) between the axis of the camera lens and the center of the screen foreground.

The constants used in this mapping process (K, a, b, Xc, and Yc) are calculated from data acquired during a calibration procedure. The calibration procedure is performed as follows:

1. video capture system **10** is activated.

2. Eight predefined points, also referred to as "standard points on the screen", are shown to the user on the computer screen, and the user is asked to touch each point, one at a time. The P and W values for each point touched by the user are recorded. Thus, at the end of the recording process 8 sets have been generated, with 4 numbers in each set (X_i, Y_i, P_i, and W_i; where "i" runs from 1 to 8 and identifies each of the eight calibration points touched on the screen, and X and Y represent the actual coordinates of the calibration points on the computer screen surface). The locations of the eight calibration points are predefined such that they cover most of the screen.

Once the above calibration procedure is complete, the software application uses the 8 calibration point data sets to calculate the values of the mapping constants (K, a, b, Xc, and Yc).

It should be noted that in the second preferred embodiment of the present invention, any object may be used as a pointer in the screen foreground, provided that the object is of identical width to the object which was used to perform the initial calibration process. So too, multiple objects may be used simultaneously, provided that they are all of the same width as the calibration object.

Focusing the dual-video tracking system on the immediate screen foreground, such that the user is able to focus on both his hand and the computer screen simultaneously, allows for the expansion of the touch-screen data input system of the current invention to include a wide spectrum of hand manipulations (other than the standard manipulation of "pushing a button") as potential activators of operational commands. Thus virtual objects depicted on the display monitor may be manipulated

by the user in a manner closely emulating natural object manipulation (the user extends his hand towards the object on the screen and then presses, pushes, pulls, or rotates the object under direct visualization, as if the object were within his grasp). This "direct" method of virtual-object manipulation simulates the usual real-life relationship between a human operator and an operational tool in a manner which is more realistic than that achievable by prior art systems, in which the virtual object on the screen is distant from the users hand, and out of his view.

There are several potential applications for the current invention:

1. The system may be used for activation of any computer software that utilizes a Graphical User Interface (GUI). Usual GUI objects (buttons, menus, scroll bars, etc.) can be activated by the user extending his finger towards the location of a GUI button on the screen. As the user's finger approaches the screen, the GUI button is pressed. As the user's finger retracts, the button is released.
2. The system may be used to activate a "zoom" function when viewing two-dimensional images on a computer screen. As the user's hand approaches the screen, the image zooms up onto that part of the 2-D image being pointed at by the user.
3. As the system tracks moving objects in the screen foreground over time, specific movement paths can be used to input operational commands to the computer. For example, as the user moves his finger along a path in the shape of an approximate circle, virtual objects on the screen can be made to rotate. Another example could be closing an application in reaction to the user moving his finger along a path that has the shape of the Latin letter "X". Furthermore, as the system can track multiple objects in the screen foreground simultaneously, complex operational commands can be input by performing simultaneous movements with two hands or fingers. For example, if the user extends both his hands towards a virtual 3D object depicted on the screen and then moves both hands one towards the other in a short rapid movement (as if closing his hands on the object), the system can understand this closing as "grabbing"of the object. The user can then "rotate" the object by moving his hands as he would if he were rotating a real life object. A similar operation (rotation) can be achieved with one hand only as the system tracks the approaching of one hand, closing of the fingers, rotation of the hand, and finally opening of the hand to signal letting go of the object.

There has therefore been described a computer touch-screen data entry system which is able to process multiple simultaneous touches, can easily be transferred from one computer display screen to another, can easily be adapted to screens of different sizes, does not degrade the quality of the display image, can sense any object which is used to touch the screen, can be used with computer screens which are not necessarily flat, and can sense additional object attributes (in addition to object location on the screen) for purposes of data input.

9. A method for entering data into a computer by interacting with a screen, comprising the steps of

a) positioning an object in a foreground of the screen;
b) acquiring a plurality of images of said screen foreground, each of said images depicting said screen foreground from a different viewpoint ;
c) processing each of said plurality of acquired images to identifya first object in said screen foreground;
d) inferring at least one descriptor of said object in each of said processed images, each of said at least one descriptor being a coordinate of a point in a virtual space; and
e) effecting a transformation of said virtual coordinates into screen coordinates describing a location of a point on the screen.

10. The method of claim 9, wherein said inferring of said unidimensional location includes defining a point of intersection between a border of said identified object and a midline of said object.

11. The method of claim 10, wherein said midline is derived by linear regression analysis.

12. The method of claim 9, wherein said descriptors include a descriptor of a width of said object and a descriptor of a spatial coordinate of said object.

13. The method of claim 12, wherein said inferring of said spatial coordinate includes defining a point of intersection between a border of said identified object and a midline of said object.

14. The method of claim 13, wherein said midline is derived by linear regression analysis.

15. The metof claim 9, wherein said transforming of said virtual coordinates into said screen coordinates includes linear interpolation.

16. The method of claim 9, wherein said transforming of said virtual coordinates into said screen coordinates includes linear extrapolation.

17. The method of claim 9, further comprising the step of

e) providing a colored background for said screen foreground,

and wherein said processing includes designating the color of at least one pixel of said image as matching said colored background.

18. The method of claim 9, further comprising the step of

e) inferring an attribute, of said identified object, selected from the group consisting of a color of said object, a spatial orientation of said object, and a size of said object.

19. The method of claim 9, further comprising the step of

e) calibrating said transformation.

20. The method of claim 19, wherein said calibration is effected by correlating a plurality of points in said virtual space with a corresponding plurality of standard points on the screen.

21. The method of claim 20, wherein for each of said standard points on the screen, said correlation is effected by,

i) positioning a second object in said screen foreground opposite said standard point on the screen;

ii) acquiring simultaneous images of said screen foreground, each of said simultaneous images depicting said screen foreground from a different viewpoint;

iii) processing each of said acquired simultaneous images to identify said second object in said screen foreground; and

iv) inferring a spatial coordinate for said identified second

object in each of said processed images; each of said spatial coordinates being a coordinate of one of said plurality of points in said virtual space.

22. The method of claim 20, wherein for each of said standard points on the screen, said correlation is effected by,

i) positioning a second object in said screen foreground opposite said standard point on the screen;

ii) acquiring a second image of said screen foreground;

iii) processing said acquired second image to identify said second object in said screen foreground; and

iv) inferring a descriptor of a width of said identified second object and a descriptor of a location of said identified second object, each of said descriptors of said second object being a coordinate of one of said plurality of points in said virtual space.

23. A system for entering data into a computer by interacting with a screen having a foreground, the system comprising:

(a) a video capture mechanism, operative to capture at least one image of the screen foreground; and

(b) an image processor, operative to identify at least one object within said at least one image, measure at least one descriptor of the at least one object, and transform the at least one descriptor into a screen coordinate.

24. A method for entering data into a computer by interacting with a screen, the method comprising the steps of:

(a) positioning an object in a foreground of the screen;

(b) acquiring at least one image of the screen foreground;

(c) processing said at least one image to identify at least one object in the screen foreground; and

(d) inferring at least one descriptor of the object, said at least one

descriptor being a coordinate of a point in a virtual space, and effecting a transformation of virtual coordinates of said virtual space into screen coordinates describing a location of a point on the screen.

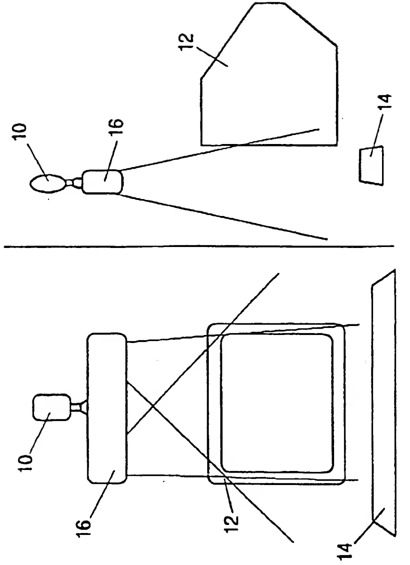


FIG. 1

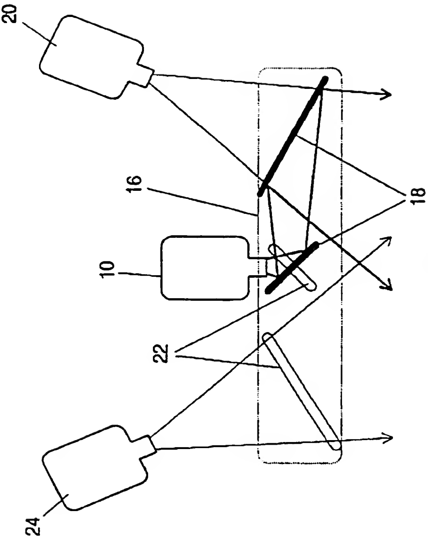


FIG. 2

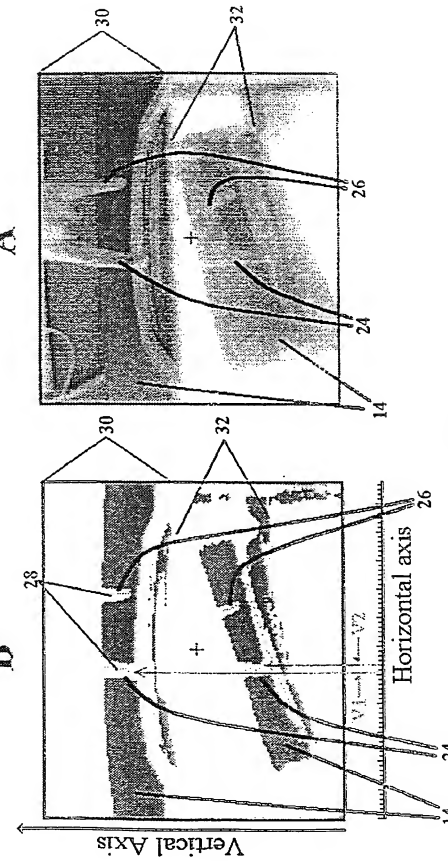


Fig. 3

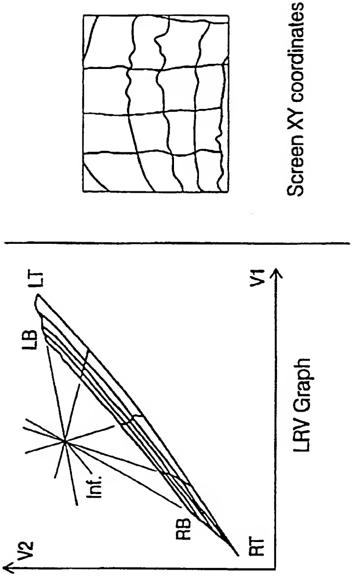


FIG. 4

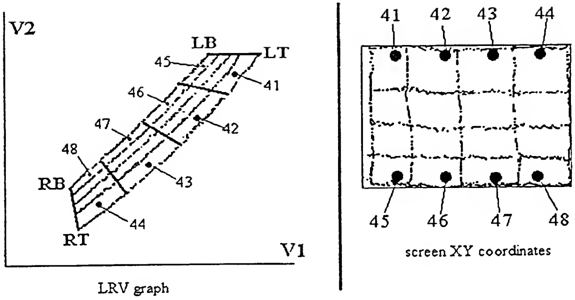
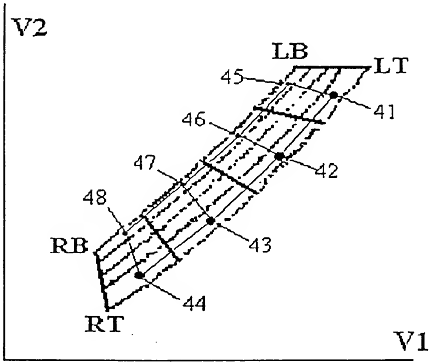

V2
LB
LT
45
41
46
47
42
48
RB
43
44
RT
V1
LRV graph
41
42
43
44
45
46
47
48
screen XY coordinates

Figure 5



LRV graph

Figure 6

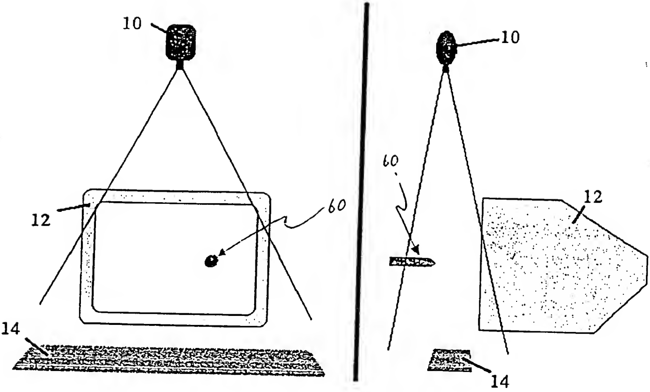
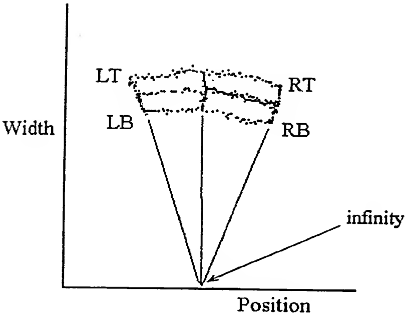


FIGURE 7



PW Graph

FIGURE 8

INTERNATIONAL SEARCH REPORT

International application No.
PCT/IL99/00083

A. CLASSIFICATION OF SUBJECT MATTER
IPC(6) : G09G 5/00
US CL : 345/114
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
U.S. : 345/114, 345/113, 345/115, 345/116, 345/133, 345/139

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
NONE

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
APS : VIDEO CAMERA, PERISCOPE, FOREGROUND, BLUE, LINEAR REGRESSION, VIRTUAL SPACE, SCREEN SPACE

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X --------- Y	US 4,672,559 A (JANSSON et al.) 09 June 1987 see col. #2, lines 1-64, Fig. 1	1-6, 9, 10, 12, 13, 18, 23, 24 . ------------- 7, 8, 11, 14-17, 19-22
Y	US 4,811,084 A (BELMARES-SARABIA et al.) 07 March 1989 see col. #1, lines 57-67	7, 8, 17

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
03 JUNE 1999	09 JUL 1999
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer for HOA BIEN Telephone No. (703) 308-6988

INTERNATIONAL SEARCH REPORT

International application No.
PCT/IL99/00083

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 5,231,674 A (CLEVELAND et al.) 27 July 1993 see col. #6, lines 57-60	11, 14
Y	US 4,810,869 A (YABE et al.) 07 March 1989 see col. #2, lines 1-8	15, 16
Y	US 5,479,597 A (FELLOUS) 26 December 1995 see col. #5, lines 1-58	19-22